Chapter 8

Austin Tesch

Date: 06-06-24

# Program 1: ch8\_pgm1;

A program with a function that accepts three arguments: an integer array, an integer size that indicates how many elements are in the array, and an integer n.

## Code:

/\*

Programmer: Austin Tesch

Date modified: 06-05-24

Compiler used:  XCODE v. 15.0

Purpose:

Create a program with a function that accepts three arguments: an integer array, an integer size that indicates how many elements are in the array, and an integer n.

The function should display all of the numbers in the array that are greater than the number n.

\*/

#include <iostream>

using namespace std;

void greaterNumbers(const int [], const int, int);

int main(){

// init array with size of 5

int const ISIZE = 5;

int array[ISIZE] {2, 4, 6, 8, 10};

// init check number

int checkInt = 5;

// check numbers in array that are greater than check number

greaterNumbers(array, ISIZE, checkInt);

return 0;

}

void greaterNumbers(const int array[], const int size, int check){

// iterate through array

for (int i = 0; i < size; i++)

// check if current element is greater than checkint

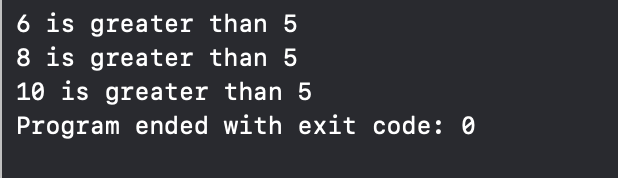
if (array[i] > check)

// number is greater

cout << array[i] << " is greater than " << check << endl;

}

## Output:



# Program 2: ch8\_pgm2;

A program that stores this information in a two-dimensional 3×7 array, where each row represents a different monkey and each column represents a different day of the week.

## Code:

/\*

Programmer: Austin Tesch

Date modified: 6/5/24

Compiler used:  XCODE v. 15.0

Purpose:

A local zoo wants to keep track of how many pounds of food each of its three monkeys eats each day during a typical week.

Write a program that stores this information in a two-dimensional 3×7 array, where each row represents a different monkey and each column represents a different day of the week.

The program should first have the user input the data for each monkey. Then it should create a report that includes the following information:

-The average amount of food eaten per day by the whole family of monkeys

-The least amount of food eaten during the week by any one monkey

-The greatest amount of food eaten during the week by any one monkey

\*/

#include <iostream>

#include <iomanip>

using namespace std;

const int MONKEYS = 3;

const int DAYS = 7;

int main() {

double food[MONKEYS][DAYS];

double totalFood = 0;

double minFood, maxFood;

// Input the food data

for (int i = 0; i < MONKEYS; ++i) {

cout << "Enter the amount of food eaten by monkey " << i + 1 << " each day of the week:\n";

for (int j = 0; j < DAYS; ++j) {

cout << "Day " << j + 1 << ": ";

cin >> food[i][j];

totalFood += food[i][j];

}

}

// Calculate average food per day

double avgFoodPerDay = totalFood / (MONKEYS \* DAYS);

// Find the minimum and maximum food eaten

minFood = maxFood = food[0][0];

for (int i = 0; i < MONKEYS; ++i) {

for (int j = 0; j < DAYS; ++j) {

// new min consumption

if (food[i][j] < minFood) {

minFood = food[i][j];

}

// new max consumption

if (food[i][j] > maxFood) {

maxFood = food[i][j];

}

}

}

// Report the results

cout << fixed << setprecision(2);

cout << "\nAverage amount of food eaten per day by the whole family of monkeys: " << avgFoodPerDay << " pounds\n";

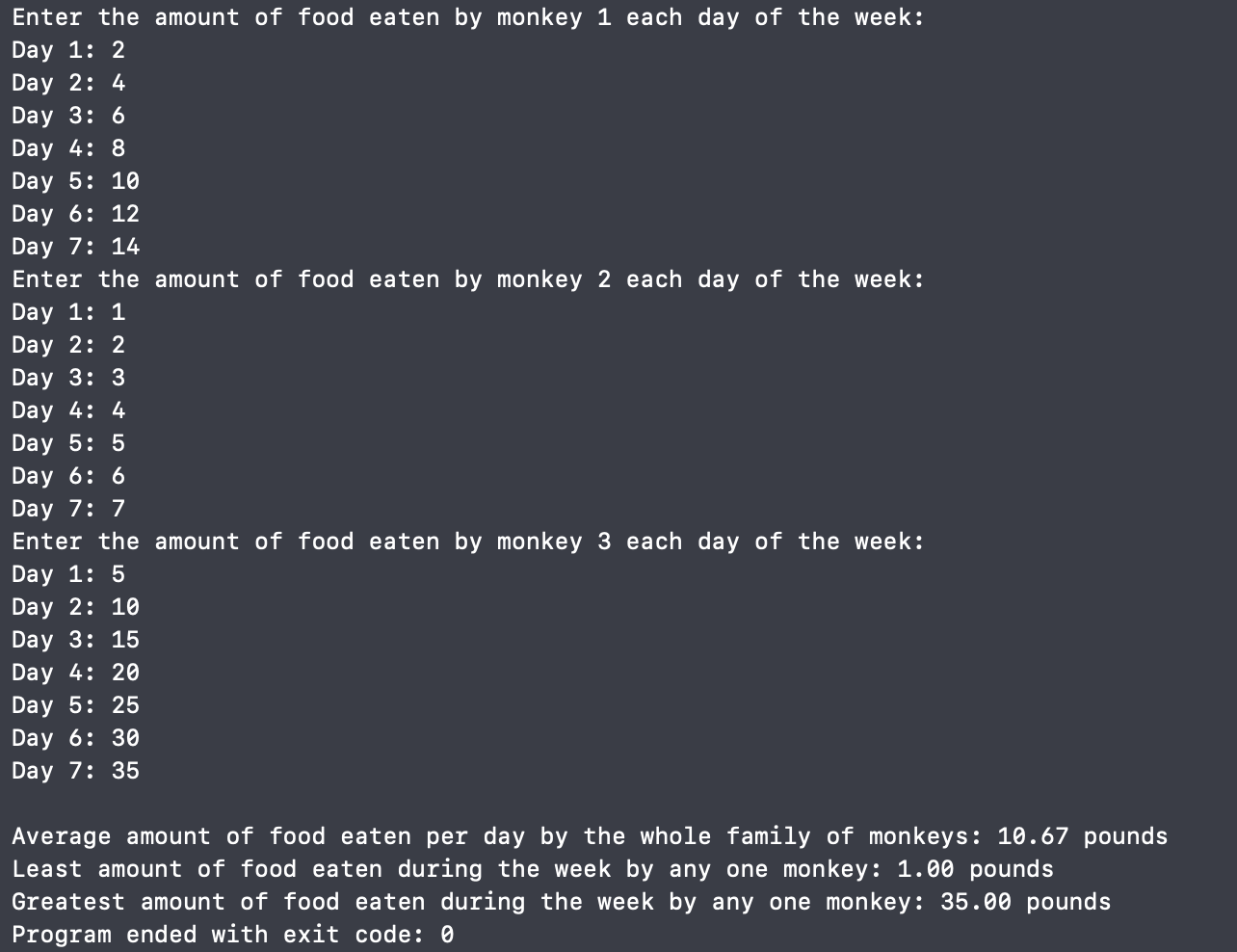
cout << "Least amount of food eaten during the week by any one monkey: " << minFood << " pounds\n";

cout << "Greatest amount of food eaten during the week by any one monkey: " << maxFood << " pounds\n";

return 0;

}

## Output:



# Program 3: ch8\_pgm3;

A simple program that uses the “CharConverter” class. It should prompt the user to input a string. Then it should call the "properWords" function and display the resulting string. Finally, it should call the uppercase function and display this resulting string. The program should loop to allow additional strings to be converted and displayed until the user chooses to quit.

## Code:

### main.cpp:

/\*

Programmer: Austin Tesch

Date modified: 6/6/24

Compiler used:  XCODE v. 15.0

Purpose:

Create a "CharConverter" class that performs various operations on strings. It should have the following two public member functions to start with.

-The "uppercase" member function accepts a string and returns a copy of it with all lowercase letters converted to uppercase.

If a character is already uppercase or is not a letter, it should be left alone.

-The "properWords" member function accepts a string of words separated by spaces and returns a copy of it with the first letter of each word converted to uppercase.

Write a simple program that uses the class. It should prompt the user to input a string. Then it should call the "properWords" function and display the resulting string. Finally, it should call the uppercase function and display this resulting string. The program should loop to allow additional strings to be converted and displayed until the user chooses to quit.

\*/

#include <iostream>

#include <string>

#include "CharConverter.hpp"

using namespace std;

int main(){

CharConverter convert;

string input\_string;

do {

// always collect at least one string

cout << "\nProvied a string for conversion or \"Q\" to quit." << endl;

getline(cin, input\_string);

if (input\_string == "Q" || input\_string == "q") {

cout << "Exiting program." << endl;

break;

}

// convert strings and output

cout << "Your string converted to all uppper case: " << convert.upperCase(input\_string) << endl;

cout << "Your string converted to \"Proper Case\": " << convert.properWords(input\_string) << endl;

} while (true); // user has quit program

return 0;

}

### CharConverter.cpp:

/\*

Programmer: Austin Tesch

Date modified: 6/6/24

Compiler used:  XCODE v. 15.0

ch8\_pgm3

\*/

#include "CharConverter.hpp"

string CharConverter::properWords(string user\_string){

// this member function converts the first letter of each word to uppercase

new\_string\_proper = user\_string;

bool new\_word = true; // first word of string will always be toupper

for (int i = 0; i < user\_string.length(); i++){ // iterate through each char in the string

if (isspace(user\_string[i])){

// space detected, new word, flag true

new\_word = true;

}

else if (new\_word){

// first char of new word toupper

new\_string\_proper[i] = toupper(user\_string[i]);

new\_word = false;

}

}

// return new string in proper word form

return new\_string\_proper;

}

string CharConverter::upperCase(string user\_string){

// this member function converts a string to all upper case letters

// clear string to prevent the last string from lingering.

new\_string\_upper = "";

for (int i = 0; i < user\_string.length(); i++){ // iterate through each char in the string

if (isalpha(user\_string[i])){

// this char is a letter

if (!isupper(user\_string[i])){

// this char is lowercase, convert to uppercase

char new\_upper = toupper(user\_string[i]);

new\_string\_upper += new\_upper;

}

else {

// this char is already upper case

new\_string\_upper += user\_string[i];

}

}

else {

// this char is a digit, space, or puncuation, leave it alone

new\_string\_upper += user\_string[i];

}

}

// return the new string with all upper case letters

return new\_string\_upper;

}

### CharConverter.hpp:

/\*

Programmer: Austin Tesch

Date modified: 6/6/24

Compiler used:  XCODE v. 15.0

\*/

#ifndef CharConverter\_hpp

#define CharConverter\_hpp

#include <string>

using namespace std;

class CharConverter{

private:

string new\_string\_upper;

string new\_string\_proper;

public:

string upperCase(string);

string properWords(string);

};

#endif /\* CharConverter\_hpp \*/

## Output:

A screenshot of a computer

Description automatically generated